Unleashing the Potential of Event-based Stereo via Coarse-to-Fine Bio-Inspired Regression

Supplementary Material

1. More Details

1.1. Event Representation

We adopt the voxel grid representation for event data following the methodology established in [9]. The implementation involves three key steps: (1) temporal normalization, where event timestamps are linearly scaled to discrete bin indices within the range $[0 : B-1]$, (2) spatial quantization, maintaining the original event data's spatial resolution ($H \times W$), and (3) volumetric aggregation, generating a 3D tensor $V_t^{L,R}(b,x,y) \in R^{B\times H\times W}$ that preserves both spatial and temporal event information. This representation can be formally expressed as:

$$V(x,y,t) = \sum_k p_k \delta(x-x_k, y-y_k) max(0, 1-\left|t-t_k^{''}\right|), \tag{1}$$

where $t_k^{''}$, t mean normalized timestamp and $t-$th temporal bin, respectively. $\delta(\cdot,\cdot)$ denotes the Dirac delta function. The discrete bins $b \in [0 : B-1]$ create a uniform temporal quantization while preserving the native H × W spatial resolution of the event camera. $t_k^{''}$ can be represented as follows:

$$t_k^{''} = \frac{B-1}{t_N - t_1}(t_k - t_1). \tag{2}$$

1.2. Feature Extraction

We adopt a three-level ResNet-like architecture. First, we downsample the input event voxel grid using three convolutions with strides of 2, 1, and 1 to obtain features at 1/2 resolution. Then, we use 16 residual layers to generate 1/4 resolution features with 64 channels, i.e., l_1. Afterwards, to expand the receptive field and enrich the semantic information, 6 residual layers are applied to obtain l_2 and l_3 with 128 channels. Finally, three feature maps with 1/4 resolution are obtained, namely l_1, l_2, and l_3. The above parameters all follow the design of GwcNet [4], considering its advanced performance. Finally, l_1, l_2, and l_3 are concatenated to form a 320-channel feature map. Then it is compressed into a 32-channel feature map through two convolutions. The above operation is performed on both the left and right events, denoted as f_L and f_R, corresponding to their feature maps, respectively, which are used to construct the cost volume.

1.3. Datasets Split

MVSEC. Following [7], we use the Indoor Flying dataset from the MVSEC dataset, which is captured from a drone flying in a room with various objects. Detailed dataset information is shown in Fig. 1.

#	Set	Sequence and frames	Size
1	Training	$S^2_{160,..,1580} \cup S^3_{125,..,1815}$	3110
	Validation	$A \in S^1_{140,..,1200}$	200
	Test	$B \in S^1_{140,..,1200} \mid A \cap B = \emptyset$	861
2	Training	$S^1_{80,..,1260} \cup S^3_{125,..,1815}$	2870
	Validation	$A \in S^2_{120,..,1420}$	200
	Test	$B \in S^2_{120,..,1420} \mid A \cap B = \emptyset$	1101
3	Training	$S^1_{80,..,1260} \cup S^2_{160,..,1580}$	2600
	Validation	$A \in S^3_{73,..,1615}$	200
	Test	$B \in S^3_{73,..,1615} \mid A \cap B = \emptyset$	1343

Figure 1. Summary of Indoor Flying splits. For each split we specify which sequences and frames are used for training and test. For example,$S^1_{140,...,1200}$ means that from sequence one only the frames 140 to 1200 are used.

interlaken	interlaken_00_a
	interlaken_00_b
	interlaken_01_a
thun	thun_01_a
	thun_01_b
zurich city	zurich_city_12_a
	zurich_city_13_a
	zurich_city_13_b
	zurich_city_14_a
	zurich_city_14_b
	zurich_city_14_c
	zurich_city_15_a

Table 1. Test sequences of the DSEC dataset.

DSEC. We use the training set provided by the benchmark for training, and upload the results of the test set to the benchmark website for testing. The training set of DSEC has 41 sequences in total, and the rest are used for testing, as shown in Table 1.

1.4. Evaluation Metrics

DSEC. Following the DSEC benchmark website, we use four standard evaluation metrics to evaluate the performance on the DSEC datasets, namely mean absolute error (MAE), root mean square error (RMSE), 1-pixel error

Method	Interlaken			Thun			Zurich City		
	MAE ↓	RMSE ↓	1PE ↓	MAE ↓	RMSE ↓	1PE ↓	MAE ↓	RMSE ↓	1PE ↓
DDES [7]	0.573	1.364	10.671	0.632	1.634	10.854	0.564	1.332	11.184
DTC-PDS [8]	0.536	1.299	9.554	0.550	1.444	8.905	0.512	1.181	9.638
Se-CFF [6]	0.514	1.207	9.377	0.553	1.483	8.506	0.517	1.192	10.075
EIS-E [5]	-	-	-	-	-	-	-	-	-
DTC-SPADE [8]	0.534	1.307	9.179	0.553	1.503	8.690	0.513	1.221	9.530
TES (3 frame) [2]	0.503	1.179	8.898	0.540	1.380	8.883	0.490	1.118	9.058
E-TAM_T-SCLM [1]	0.496	1.125	8.984	0.525	1.335	8.567	0.496	1.109	9.587
TES (4 frame) [2]	0.496	1.184	8.589	0.527	1.377	8.437	0.481	1.106	8.796
C2F-HUMAN (Ours)	**0.458**	**1.011**	**8.100**	**0.496**	**1.220**	**8.143**	**0.456**	**0.997**	**8.404**

Table 2. Evaluation results for each sequence on the DSEC [3] dataset. The best is in **bold** and the second best is in underline. - indicates that results are not provided in the original paper. The results have been released on the DSEC disparity benchmark website[1].

(1PE), and 2-pixel error (2PE). Specifically, MAE measures the average absolute difference between predicted disparity d_p and ground truth d_{gt} over all valid pixels N:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|d_p^{(i)} - d_{gt}^{(i)}\right|. \tag{3}$$

1PE calculates the percentage of pixels where the absolute disparity error exceeds one pixel:

$$1PE = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\left(\left|d_p^{(i)} - d_{gt}^{(i)}\right| > 1\right) \times 100. \tag{4}$$

2PE is calculated similarly. RMSE calculates the root mean square error of the disparity, penalizing larger errors quadratically. The formula is as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(d_p^{(i)} - d_{gt}^{(i)}\right)^2}. \tag{5}$$

MVSEC. we use four evaluation metrics to evaluate the performance on the MVSEC dataset, namely mean disparity error, one-pixel accuracy, mean depth error, and median depth error. The mean disparity error measures the average absolute difference between predicted disparity and ground truth disparity over all valid pixels N: $\frac{1}{N}\sum_{i=1}^{N}\left|d_p^{(i)} - d_{gt}^{(i)}\right|$. One-pixel accuracy calculates the percentage of pixels with absolute disparity error ≤ 1 pixel: $\frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\left(|| \leq 1\right) \times 100$ Mean depth error calculates the average absolute error between predicted depth and ground truth depth , measured over all valid pixels N: $\frac{1}{N}\sum_{i=1}^{N}\left|z_p^{(i)} - z_{gt}^{(i)}\right|, z = \frac{fB}{d}$. The median depth error represents the middle value of all per-pixel errors: $median\left\{\left|z_p^{(i)} - z_{gt}^{(i)}\right|\right\}_{i=1}^{N}$.

2. More Quantitative Results

2.1. More Detailed Quantitative Results of DSEC

The DSEC dataset offers high-resolution stereo event data captured in diverse outdoor driving scenarios, comprising 53 sequences under varying illumination conditions and dynamic velocities. Following the dataset's standard protocol, we utilize 41 sequences for training and evaluate on the three designated test sequences—Interlaken, Zürich City, and Thun—with aggregated results reported in the main paper and per-sequence breakdowns provided in Table 2. Our method achieves state-of-the-art performance across all test sequences, demonstrating statistically significant improvements over existing event-based stereo matching approaches. Notably, as the ground truth disparity for the DSEC test set is not publicly available, quantitative comparisons are conducted via the official benchmark website[1], from which we compile competing methods' results for fair evaluation. Specifically, the proposed method outperforms the second best metric by 7.66%, 5.52%, and 5.19% in MAE, 10.13%, 8.61%, and 9.85% in RMSE, and 5.69%, 3.48%, and 4.45% in 1PE on interlaken, thun, and zurich city, respectively.

3. More Qualitative Results

3.1. Ablation of R&EC

Although the disparity level information output by the coarse regression stage can provide prior information for subsequent fine regression, it may contain errors. The proposed R&EC module is designed to correct for errors that may occur in the coarse regression. Therefore, we visualize the R&EC module's ablation experiments to visually verify its effectiveness, as shown in Fig. 2. As can be seen, the addition of R&EC effectively corrects potential errors in the coarse regression, ensuring the accuracy of the prior

[1] https://dsec.ifi.uzh.ch/uzh/disparity-benchmark/

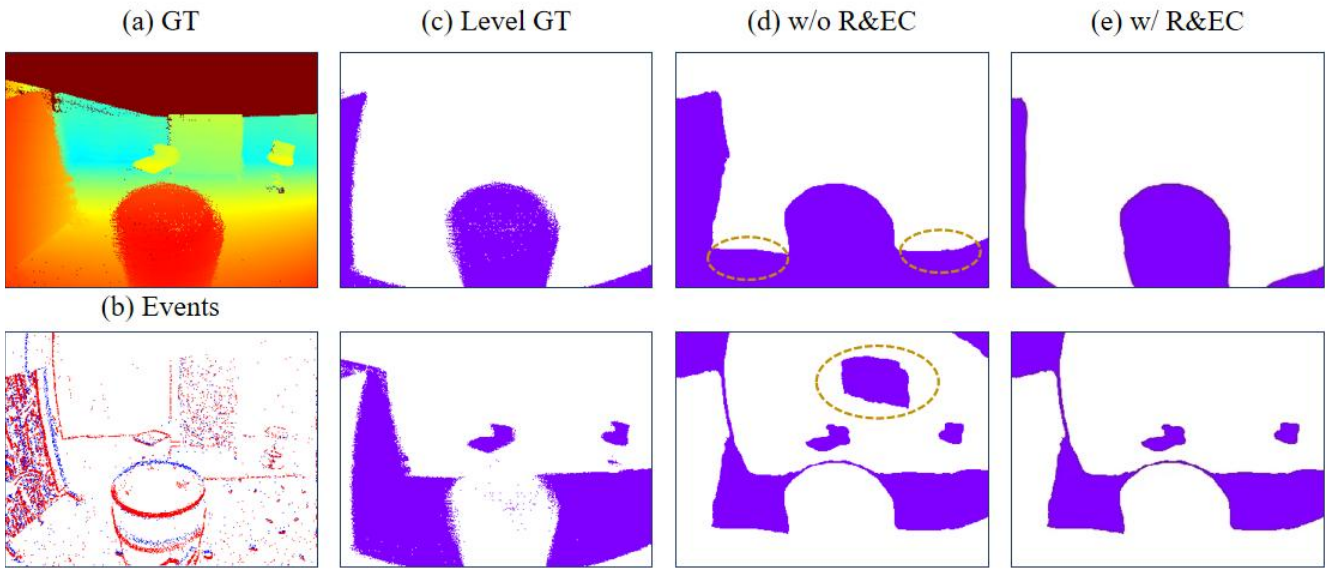

Figure 2. Visual comparison with and without R&EC. The yellow dotted circle marks the disparity level error that occurs without R&EC.

information it provides.

4. Discussions

4.1. Significance of Our Approach

This work proposes a coarse-to-fine regression framework inspired by the human visual system to address the challenges posed by the wide range of disparity labels. Through qualitative and quantitative experiments, we demonstrate the effectiveness of the proposed components. We innovatively divide the disparity regression module into two parts. The coarse regression classifies pixels into disparity levels, providing a prior for the subsequent fine regression, making accurate disparity regression much easier. The value of this work lies in its exploration of a specific mechanism in the human visual system. Our insights encourage the community to adopt more mechanisms from the visual system for various vision tasks.

4.2. Why Use Events

Event cameras exhibit unique hardware characteristics, such as microsecond-level latency ($<$ 1 ms) and a high dynamic range (HDR) exceeding 120 dB, enabling robust performance in challenging illumination conditions (e.g., low-light or high-contrast scenarios) and high-speed motion environments. These neuromorphic sensors operate asynchronously, capturing per-pixel brightness changes (log-intensity gradients) at temporal frequencies up to 1 MHz, thereby preserving fine-grained motion dynamics and edge features often lost in frame-based systems due to motion blur or sampling limitations. By complementing traditional frame-based cameras, event cameras address critical failure modes of conventional vision sensors—such as under rapid motion or extreme lighting variations—while providing sparse but high-temporal-precision data conducive to efficient downstream processing.

4.3. Future Extension

Our work presents the first successful integration of the neural mechanisms of disparity processing in the visual cortex into event-based stereo matching. The effectiveness of our simple yet biologically-inspired operation is not only encouraging but also opens up a new avenue for community. Our findings pave the way for a new research direction: the systematic translation of well-established neurobiological findings into efficient model components We believe this serves as a compelling proof-of-concept, suggesting that the human visual system remains a rich source of inspiration for building more robust and efficient models. Future work will focus on exploring other cortical mechanisms for a wider range of event-based vision tasks.

4.4. Limitation

While our method performs well in experiments and is inspired by biology, it still has some limitations that open promising directions for future research. The proposed disparity-level discretization effectively narrows the hypothesis space for fine-grained regression. However, the visual cortex processes disparity through more than one mechanism; there may be other, more complex mechanisms at play. In summary, these limitations highlight the opportunity to extend biologically inspired hierarchical processing methods to the realm of event-based perception, offering new possibilities for achieving robust, efficient, and scal-

able stereo matching.

References

[1] Wu Chen, Yueyi Zhang, Xiaoyan Sun, and Feng Wu. Event-based stereo depth estimation by temporal-spatial context learning. *IEEE Signal Processing Letters*, 31:1429–1433, 2024. 2

[2] Hoonhee Cho, Jae-Young Kang, and Kuk-Jin Yoon. Temporal event stereo via joint learning with stereoscopic flow. In *Proceedings of the European Conference on Computer Vision*, pages 294–314. Springer, 2025. 2

[3] Mathias Gehrig, Willem Aarents, Daniel Gehrig, and Davide Scaramuzza. Dsec: A stereo event camera dataset for driving scenarios. *IEEE Robotics and Automation Letters*, 6(3):4947–4954, 2021. 2

[4] Xiaoyang Guo, Kai Yang, Wukui Yang, Xiaogang Wang, and Hongsheng Li. Group-wise correlation stereo network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3273–3282, 2019. 1

[5] Mohammad Mostafavi, Kuk-Jin Yoon, and Jonghyun Choi. Event-intensity stereo: Estimating depth by the best of both worlds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4258–4267, 2021. 2

[6] Yeongwoo Nam, Mohammad Mostafavi, Kuk-Jin Yoon, and Jonghyun Choi. Stereo depth from events cameras: Concentrate and focus on the future. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6114–6123, 2022. 2

[7] Stepan Tulyakov, Francois Fleuret, Martin Kiefel, Peter Gehler, and Michael Hirsch. Learning an event sequence embedding for dense event-based deep stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1527–1537, 2019. 1, 2

[8] Kaixuan Zhang, Kaiwei Che, Jianguo Zhang, Jie Cheng, Ziyang Zhang, Qinghai Guo, and Luziwei Leng. Discrete time convolution for fast event-based stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8676–8686, 2022. 2

[9] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Unsupervised event-based learning of optical flow, depth, and egomotion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 989–997, 2019. 1